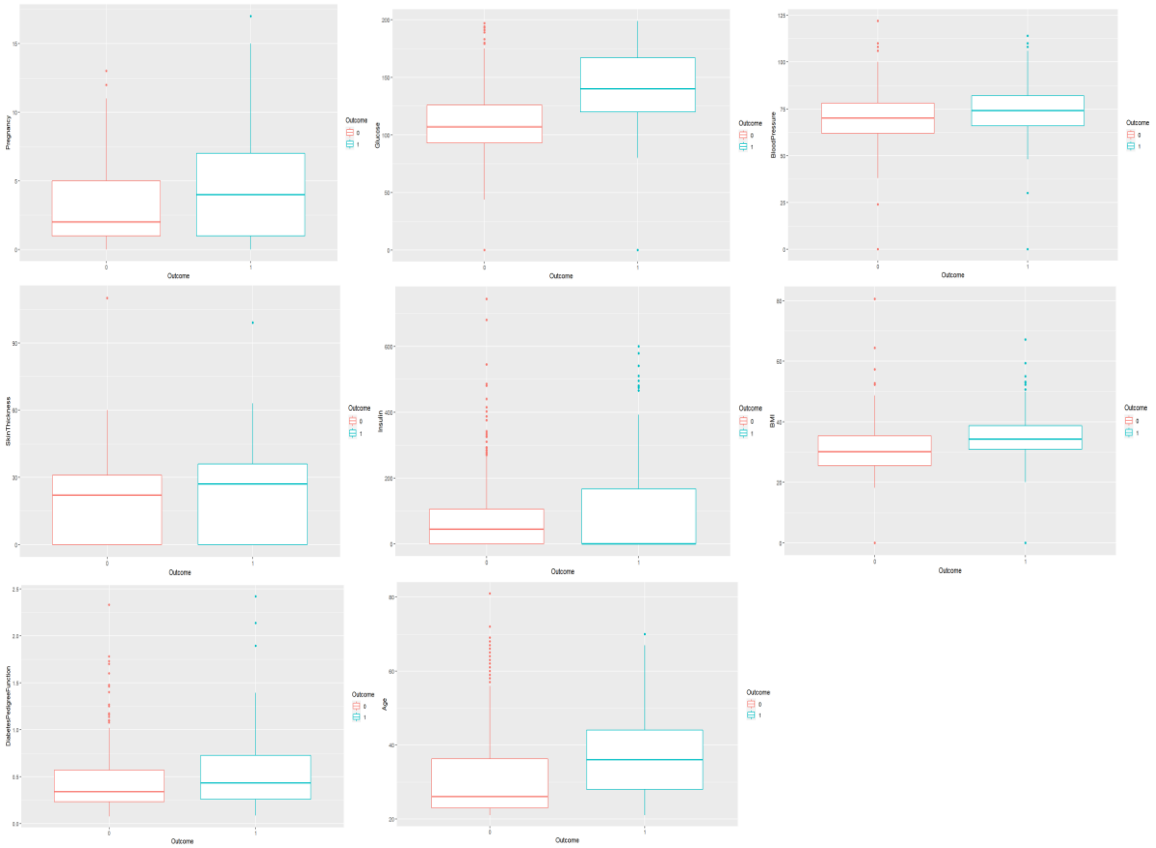
STAT 4360 (Introduction to Statistical Learning, Fall 2022)
Mini Project 3

Name: Haeun Kim

1.(a)



- We set the Outcome as the reponse, the other variables as predictors. So, I want to figure out the relationship between response and predictors. And I used boxplot due to response is categorical. This is the boxplot for each Predictors versus Outcome. 0 means no diabetes, and 1 means diabetes from the Outcome. I could find that the all the predictors have some positive correlations with Outcome. Except for Insuline, all predictors' mean increases. In case of Insuline, although the mean decreases, but the 75th percentile increases, and maximum observation below upper fence also increases. So, I could think that all the predictors have some positive correlations with Outcome.

(b)

- I performed logistic regression by using all the predictors and Outcome as the response to test hypothesis testing to find the "reasonably good" logistic refession model. When I see tha p-value for this generalized linear model, p-value for the SkinThickness has high p-value. I can reject that the SkinThickness have significant relationship to Ouncome. So, I performed logistic regression by using all the predictors (except for the SkinThickness), and it seems that all the predictors have significant relationship to response. When comparing the full model and reduced model by using ANOVA by checking the Pr(>Chi), I could conclude that the dropped predictor (SkinThickness) is not significant. When comparing the reduced model and the null model by using ANOVA by checking the Pr(>Chi), I

could conclude that the all the predictors from reduced model are significant because its p-value is < 2.2e-16. So, I can conclude that the reduced model is “reasonably good” logistic regression model for these data

```
Call:
glm(formula = Outcome ~ Pregnancies.. + Glucose.. + BloodPressure.. +
    SkinThickness.. + Insulin.. + BMI.. + DiabetesPedigreeFunction.. +
    Age.., family = binomial, data = diabetes)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.1942  -0.7256  -0.4473   0.7540   2.8979

Coefficients:
                           Estimate Std. Error z value Pr(>|z|)
(Intercept)              -8.0264511  0.4306345 -18.639  < 2e-16 ***
Pregnancies..             0.1263845  0.0199997   6.319 2.63e-10 ***
Glucose..                 0.0337202  0.0022258  15.150  < 2e-16 ***
BloodPressure..          -0.0096446  0.0032441  -2.973  0.00295 **
SkinThickness..           0.0005185  0.0042301   0.123  0.90244
Insulin..                -0.0012426  0.0005786  -2.148  0.03175 *
BMI..                     0.0775549  0.0088819   8.732  < 2e-16 ***
DiabetesPedigreeFunction..  0.8877583  0.1860275   4.772 1.82e-06 ***
Age..                     0.0129414  0.0057020   2.270  0.02323 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> anova(fit2, fit1, test = "Chisq")
Analysis of Deviance Table

Model 1: Outcome ~ Pregnancies.. + Glucose.. + BloodPressure.. + Insulin.. +
    BMI.. + DiabetesPedigreeFunction.. + Age..
Model 2: Outcome ~ Pregnancies.. + Glucose.. + BloodPressure.. + SkinThickness.. +
    Insulin.. + BMI.. + DiabetesPedigreeFunction.. + Age..
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1      1992     1914.3
2      1991     1914.3  1 0.015033   0.9024
```

```
Call:
glm(formula = Outcome ~ Pregnancies.. + Glucose.. + BloodPressure.. +
    Insulin.. + BMI.. + DiabetesPedigreeFunction.. + Age.., family = binomial,
    data = diabetes)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2028  -0.7253  -0.4454   0.7557   2.8980

Coefficients:
                           Estimate Std. Error z value Pr(>|z|)
(Intercept)              -8.0273146  0.4306244 -18.641  < 2e-16 ***
Pregnancies..             0.1263707  0.0199944   6.320 2.61e-10 ***
Glucose..                 0.0336810  0.0022020  15.296  < 2e-16 ***
BloodPressure..          -0.0095806  0.0032013  -2.993  0.00276 **
Insulin..                -0.0012123  0.0005228  -2.319  0.02042 *
BMI..                     0.0778743  0.0084946   9.167  < 2e-16 ***
DiabetesPedigreeFunction..  0.8894946  0.1855205   4.795 1.63e-06 ***
Age..                     0.0128944  0.0056879   2.267  0.02339 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> fit3 <- glm(Outcome ~ 1, family = binomial, data = diabetes)
> anova(fit3, fit2, test = "Chisq")
Analysis of Deviance Table

Model 1: Outcome ~ 1
Model 2: Outcome ~ Pregnancies.. + Glucose.. + BloodPressure.. + Insulin.. +
    BMI.. + DiabetesPedigreeFunction.. + Age..
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1      1999     2569.4
2      1992     1914.3  7   655.06 < 2.2e-16 ***
```

(c)

→ The final model

- Y(Outcome) = -8.0273146 + 0.1263707*b1 + 0.0336810*b2 – 0.0095806*b3 – 0.0012123*b4 + 0.0778743*b5 + 0.8894946*b6 + 0.0128944*b7

(b1= Pregnancies, b2 = Glucose, b3 = BloodPressure, b4 = Insulin, b5 = BMI, b6 = DiabetesPredigreeFuncion, b7 = Age)

The Estimates of regression coefficients and Standard error of the estimates

	Estimates of regression coeff	Standard error	2.5%	97.5%
(Intercept)	-8.0273146	0.4306244	-8.889630784	-7.2009252668
Pregnancy	0.1263707	0.0199944	0.087447559	0.1658700222
Glucose	0.0336810	0.0022020	0.029435255	0.0380709843
BloodPressure	-0.0095806	0.0032013	-0.015885768	-0.0033221648
Insulin	-0.0012123	0.0005228	-0.002241105	-0.0001893038
BMI	0.0778743	0.0084946	0.061474284	0.0947952879
DiabetesPredigreeFunction	0.8894946	0.1855205	0.527470753	1.2549028449
Age	0.0128944	0.0056879	0.001711033	0.0240290378

Above table shows us the estimates of the regression coefficients, the standard errors of the estimates, and 95% confidence interval. The logistic regression coefficient 0.1263707 associated with Pregnancy is the expected change(increase) in log odds of having the Outcome(diabetes) per unit change in Pregnancy. The logistic regression coefficient 0.0336810 associated with Glucose is the expected change(increase) in log odds of having the Outcome(diabetes) per unit change in Glucose. The logistic regression coefficient -0.0095806 associated with BloodPressure is the expected change(decrease) in log odds of having the Outcome(diabetes) per unit change in Glucose. The train error rate is 0.216 (=21.6%)

2.(a) - confusion matrix

lr.pred	0	1
0	1180	296
1	136	388

- The error rate for a fitted logistic regression model using all predictors in the data 0.216(=21.6%).

The specificity for a fitted logistic regression model using all predictors in the data is 0.8966565(= 89.6%)

The sensitivity for a fitted logistic regression model using all predictors in the data is 0.5672515 (=56.7%)

(b)

- The test error rate of the model that I estimated by using my own code is 0.2195(=21.95%).

(c)

- The accuracy of the model that I got by using "caret" is 0.7805(=78.05%). So, the test error rate of the model is 1-0.7805 = 0.2195(=21.95%), and I got the same result from (b).

(d)

- The accuracy of the model that I got by using "caret" and my own function if 0.7815(=78.15%). So, the test error rate of the model is 1-0.7815 = 0.2185(=21.85%)

(e)

- I was a little confused which model should I use. So I figured out both of full model and proposed logistic regression from Problem 1.

- (Full model) The Accuracy of the model by using both LDA function and the caret is 0.777(=77.7%). So, the test error rate is 1-0.777 = 0.223(=22.3%).

-(Proposed model) The Accuracy of the model by using both LDA function and the caret is 0.781(=78.1%). So, the test error rate is 1-0.781 = 0.219(=21.9%).

(f)

- I was a little confused which model should I use. So I figured out both of full model and proposed logistic regression from Problem 1.

- (Full model) The Accuracy of the model by using both QDA function and the caret is 0.7555 (=75.55%). So, the test error rate is 1-0.7555 = 0.2445(=24.45%).

-(Proposed model) The Accuracy of the model by using both QDA function and the caret is 0.7625 (=76.25%). So, the test error rate is 1-0.7625 = 0.2375(=23.75%).

(g)

- To find out the optimal K, I used the tune.knn function, and I set k from 1 to 50 because it takes to much time if I set the large range of K.

- And the optimal K that I got is "1".

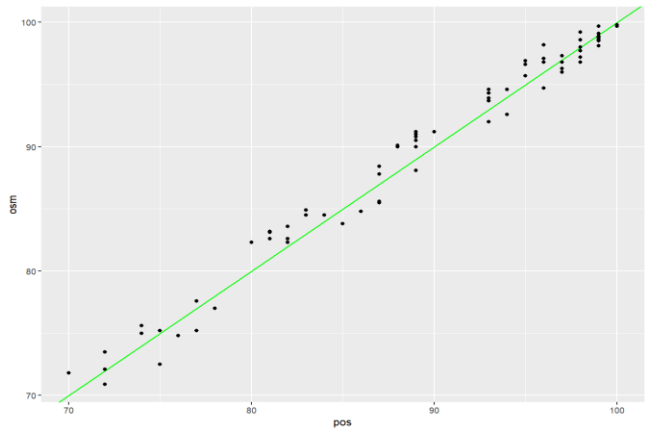
- (Full model) The Accuracy of the model by the caret for KNN is 0.9985(=99.85%). So, the test error rate is 1-0.9985 = 0.0015(=0.15%).

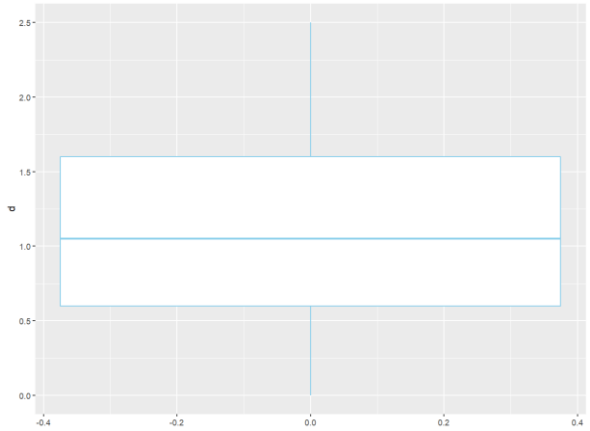
-(Proposed model) The Accuracy of the model by the caret for KNN is 0.999 (=99.9%). So, the test error rate is 1-0.999 = 0.001(=0.1%).

(h)

- By results from the logistic regression, LDA, QDA, and KNN, I would recommend KNN. This is because it has the least test error rate for both full model and proposed model(Reduced model) among the classifiers.

3.(a)





- The majority of points of the scatter plot between two method are not on the 45-degree line, however, the points are near the 45-degree line, and those points would seem like making the 45-degree line because those are close to the 45-degree line. So, I could think that two methods have a good areement. When looking at the difference of two method box plot, the median is a little bit greater than 1.0 and the lower quantile is around 0.6, and the upper quantile is around 1.7. I could think that these two methods have a good areement.

(b)

- I think the smaller values for theta imply better agreement. As the difference of two methods is small, the points have the large possibility to be on or near the 45-degree line, which implies better agreement. This is because the methods would have prefect agreement if all the points in the scatterplot fell on the 45-degree line, or equivalently, all the differences were zero. Total deviation index (TDI) means “quantile of the allowable coverage probability based on absolute difference). So, if the TDI is small, the observations by the two methods could imply better agreement.

(c)

- I got 2 as the theta_hat with 0.9 probability.

(d)

- I got bias as 0.00207, which is the difference between the mean of the bootstrap estimates of θ and the sample estimate of θ, and 0.07100496 as stadard error. I got the 95% confidence interval (1.78, 2.20). So, 95% upper confidence bound for theta is 2.20. With 95% of the times, this bootstrap method accurately results in a confidence interval (1.78, 2.20).

(e)

- I got bias as 0.00476, which is the difference between the mean of the bootstrap estimates of θ and the sample estimate of θ, and 0.1257742 as stadard error. I got the 95% confidence interval (1.78, 2.20). So, 95% upper confidence bound for theta is 2.20. With 95% of the times, this bootstrap method accurately results in a confidence interval (1.78, 2.20).

- there is no big difference between biases, but both of biases are quite small, but there is a little difference between Standard errors, but both of them have the same Confidence Intervel with (1.78, 2.20).

(f)

- I would say that the methods agree well enough to be used interchangeably in practice. This is because the bias and standard error is low. In addition, with 95% confidence interval, 0.9th quantile of |D| have high possibility is between (1.78, 2.20). So, I agree these two methods have a good areement.

R Code

```
library(e1071)
library(ggplot2)
library(ISLR2)
library(caret)
library(boot)
setwd("C:/Users/haeun/OneDrive/문서/STAT33550")

#Bringing the oxyge dataset
oxygen <- read.table("oxygen_saturation.txt", header = TRUE)
#Bringing the diabetes dataset
diabetes <- read.csv("diabetes.csv", header = TRUE)

#Factoring the Outcome from the diabetes dataset
diabetes$Outcome <- as.factor(diabetes$Outcome)
# make the data set for training dataset X
train.y <- diabetes$Outcome
# make the data set for training dataset X
train.x <- diabetes[,-9]

#Question 1-(a)
# Make the boxplot to figure out the predictors to response(Outcome)
ggplot(data = diabetes, aes(x = Outcome, y = Pregnancies.., color =Outcome ))+
  geom_boxplot() +
  labs(x = "Outcome", y = "Pregnancy")
ggplot(data = diabetes, aes(x = Outcome, y = Glucose.., color = Outcome))+
  geom_boxplot()+
  labs(x = "Outcome", y = "Glucose")
ggplot(data = diabetes, aes(x = Outcome, y = BloodPressure.., color = Outcome))+
  geom_boxplot() +
  labs(x = "Outcome", y = "BloodPressure")
ggplot(data = diabetes, aes(x = Outcome, y = SkinThickness.., color = Outcome))+
  geom_boxplot()+
  labs(x = "Outcome", y = "SkinThickness")
ggplot(data = diabetes, aes(x = Outcome, y = Insulin.., color = Outcome))+
  geom_boxplot()+
  labs(x = "Outcome", y = "Insulin")
ggplot(data = diabetes, aes(x = Outcome, y = BMI.., color = Outcome))+
  geom_boxplot()+
  labs(x = "Outcome", y = "BMI")
ggplot(data = diabetes, aes(x = Outcome, y = DiabetesPedigreeFunction.., color = Outcome))+
  geom_boxplot()+
  labs(x = "Outcome", y = "DiabetesPedigreeFunction")
```

```
ggplot(data = diabetes, aes(x = Outcome, y = Age.., color = Outcome))+
  geom_boxplot()+
  labs(x = "Outcome", y = "Age")

#Question1 - (b)
#Factoring the Outcome from the diabetes dataset
diabetes$Outcome <- as.factor(diabetes$Outcome)
fit1 <- glm(Outcome ~ Pregnancies.. + Glucose.. + BloodPressure.. + SkinThickness.. +
            Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age.., family = binomial, data =
diabetes)
summary(fit1)
#Logistic Regression after dropping the SkinThickness
fit2 <- glm(Outcome ~ Pregnancies.. + Glucose.. + BloodPressure.. +
            Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age.., family = binomial, data =
diabetes)
summary(fit2)
#ANOVA for testing the significance of dropped predictor
anova(fit2, fit1, test = "Chisq")
# The logistic regressiong for Null model
fit3 <- glm(Outcome ~ 1, family = binomial, data = diabetes)
summary(fit3)
#ANOVA for testing the significance of all predictors from the reduced model
anova(fit3, fit2, test = "Chisq")

#Question 1-(c)
#For the 95% confidence interval
confint(fit2, level = 0.95)
# the train dataset
diabetes.train <- subset(diabetes)
# Estimated probabilities for train data
lr.prob <- predict(fit2, diabetes.train, type = "response")
# Predicted classes (using 0.5 cutoff)
lr.pred <- ifelse(lr.prob >= 0.5, "1", "0")
# Train error rate
train_error_rate <- 1 - mean(lr.pred == diabetes.train$Outcome)

#Question 2-(a)
# Estimated probabilities for train data
lr.prob <- predict(fit1, diabetes, type = "response")
# Predicted classes (using 0.5 cutoff)
lr.pred <- ifelse(lr.prob >= 0.5, "1", "0")
# Maing confusion matrix
confusion1 <- table(lr.pred, diabetes$Outcome)
print(confusion1)
```

```
# Calaulate the error rate
error_rate <- (296+136)/(1180+296+136+388)
# Calculate the sensitivity
sensitivity <- 388/(296+388)
# Calculate the specificity
specificity <- 1180/(1180+136)
#Question 2-(b)
# number of row of dataset for LOOCV
n <- nrow(diabetes)
cv.err <- sapply(1:n, FUN = function(i){
  fit <- glm(Outcome~Pregnancies.. + Glucose.. + BloodPressure.. + SkinThickness.. +
               Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age..,, data = diabetes[-i,], family =
binomial)
  one_predic <- predict(fit, diabetes[i, ], type = "response") >= 0.5
  one_predic <- ifelse(one_predic, "1", "0")

  one_predic != diabetes$Outcome[i]
})
mean(cv.err)
#Question 2-(c)
#Using the caret library to calculate the LOOCV error rate
ctrl <- trainControl(
  method = 'LOOCV',
  number = 1
  )
loocv_caret <- train(Outcome~Pregnancies.. + Glucose.. + BloodPressure.. + SkinThickness.. +
                       Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age.., data = diabetes, method =
"glm", trControl = ctrl)
loocv_caret$result
#Question 2-(d)
#Using the caret library to calculate the LOOCV error rate
ctrl <- trainControl(
  method = 'LOOCV',
  number = 1
)
loocv_caret <- train(Outcome~Pregnancies.. + Glucose.. + BloodPressure.. +
                       Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age.., data = diabetes, method =
"glm", trControl = ctrl)
loocv_caret$result
#Question 2-(e)
#Performing LDA for diabetes dataset for Full model
acc <- NULL
lda_dia <- for(i in 1:2000){
  fit <- lda(Outcome~Pregnancies.. + Glucose.. + BloodPressure.. + SkinThickness.. +
```

```
          Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age..,, data = diabetes[-i,])
  one_predic <- predict(fit, diabetes[i, ])$class
  acc = c(acc, one_predic == diabetes$Outcome[i])
}
mean(acc)

#LDA on LOOCV by using the caret package
loocv_caret_e <- train(Outcome~Pregnancies.. + Glucose.. + BloodPressure.. + SkinThickness.. +
                  Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age.., data = diabetes, method =
"lda", trControl = ctrl)
loocv_caret_e$result
#Performing LDA for diabetes dataset for proposed(Reduced) model
acc <- NULL
# By using the lda function and LOOCV to figure out the accuracy
lda_dia <- for(i in 1:2000){
  fit <- lda(Outcome~Pregnancies.. + Glucose.. + BloodPressure.. +
          Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age..,, data = diabetes[-i,])
  one_predic <- predict(fit, diabetes[i, ])$class
  acc = c(acc, one_predic == diabetes$Outcome[i])
}
#Accuracy of LDA from proposed(Reduced) model
mean(acc)

#LDA on LOOCV by using the caret package
loocv_caret_e <- train(Outcome~Pregnancies.. + Glucose.. + BloodPressure.. +
                     Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age.., data = diabetes, method
= "lda", trControl = ctrl)
loocv_caret_e$results
#Question 2-(f)
#Performing QDA for diabetes dataset for Full model
acc <- NULL
# By using the qda function and LOOCV to figure out the accuracy
qda_dia <- for(i in 1:2000){
  fit <- qda(Outcome~Pregnancies.. + Glucose.. + BloodPressure.. + SkinThickness.. +
          Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age..,, data = diabetes[-i,])
  one_predic <- predict(fit, diabetes[i, ])$class
  acc = c(acc, one_predic == diabetes$Outcome[i])
}
#Accuracy of QDA from full model
mean(acc)
#QDA on LOOCV by using the caret package
loocv_caret_e <- train(Outcome~Pregnancies.. + Glucose.. + BloodPressure.. + SkinThickness.. +
                  Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age.., data = diabetes, method =
"qda", trControl = ctrl)
```

```
loocv_caret_e$result
#Performing QDA for diabetes dataset for proposed(Reduced) model
acc <- NULL
# By using the lda function and LOOCV to figure out the accuracy
qda_dia <- for(i in 1:2000){
  fit <- qda(Outcome~Pregnancies.. + Glucose.. + BloodPressure.. +
           Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age..,, data = diabetes[-i,])
  one_predic <- predict(fit, diabetes[i, ])$class
  acc = c(acc, one_predic == diabetes$Outcome[i])
}
#Accuracy of QDA from proposed(Reduced) model
mean(acc)
#QDA on LOOCV by using the caret package
loocv_caret_e <- train(Outcome~Pregnancies.. + Glucose.. + BloodPressure.. +
                  Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age.., data = diabetes, method =
"qda", trControl = ctrl)
loocv_caret_e$result
#Question 2-(g)
#To find out the Optimal K
knn.cross <- tune.knn(x = train.x, y = train.y, k = 1:50,tunecontrol=tune.control(cross=2000))
summary(knn.cross)
optimal_K <- 1
#KNN for the full model
loocv_caret_g <- train(Outcome~Pregnancies.. + Glucose.. + BloodPressure.. + SkinThickness.. +
                  Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age.., data = diabetes, method =
"knn",tuneGrid = data.frame(k = 1), metric = "Accuracy", trControl = ctrl)
loocv_caret_g$result
#KNN for the proposed(Reduced) model
loocv_caret_g <- train(Outcome~Pregnancies.. + Glucose.. + BloodPressure.. +
                  Insulin..  + BMI.. + DiabetesPedigreeFunction.. + Age.., data = diabetes, method =
"knn",tuneGrid = data.frame(k = 1), metric = "Accuracy", trControl = ctrl)
loocv_caret_g$result
############################################################
#Question 3-(a)
# The scatter plot with superimposing the 45 degree
 ggplot(oxygen, aes(x = pos, y = osm))+
  geom_abline(intercept = 0, slope = 1, size = 0.5, color = "Green", labs =) +
  geom_point()
# absolute values of differences in the measurements from the two methods.
oxygen$d <-abs(oxygen$pos-oxygen$osm)
# The boxplot with absolute values of differences in the measurements from the two methods.
ggplot(oxygen, aes(y=d))+
  geom_boxplot(color = "skyblue")
#Question 3-(c)
```

```
theta_hat <- quantile(oxygen$d, probs = c(0.9))
#Question 3-(d)
boot.result <- numeric(1000)
set.seed(1)
for(i in 1:1000){
  boot.samp <- sample(oxygen$d, length(oxygen$d), replace=TRUE)
  boot.result[i] <- boot.fn(boot.samp)
  }
flat <- mean(boot.result)
bias <- theta - theta_hat
bias
se <- mean(replicate(1000, sd(boot.samp)/sqrt(length(oxygen$d))))
confidence <- quantile(boot.result, c(0.025, 0.975))
#Question 3-(e)
boot.fn <- function(oxygen, index) return(quantile(oxygen[index], probs = c(0.9)))
set.seed(1)
boot <- boot(oxygen$d, boot.fn, 1000)
boot
boot.ci(boot.out=boot)
```